CLAIMS

What is claimed is:

1. A method for static single assignment form dead code elimination, the method comprising:

examining a first instruction off of a worklist, wherein the first instruction includes a previous link and a write mask;

examining at least one second instruction of the worklist, wherein the at least one second instructions are sources of the first instruction and each of the at least one second instructions include a previous link and a write mask;

determining if any components within a particular field are required for the at least one second instruction; and

if no components are live, delete the first instruction from the machine code.

2. The method of claim 1 further comprising:

generating the worklist by:

for each of a plurality of instructions, determining if the instruction is a critical instruction; and

if the instruction is a critical instruction, writing the instruction to the worklist.

3. The method of claim 2 further comprising:

setting a live bit for each component of the plurality of instructions.

4. The method of claim 2 wherein the critical instruction is an instruction that generates an export value.

5. The method of claim 2 further comprising:

prior to generating the worklist:

receiving a plurality of instructions;

adding to each instruction a previous link; and

adding to each instruction a write mask.

6. The method of claim 5 wherein the write mask is a multi-bit field representing a number of components in a superword register.

7. The method of claim 6 wherein each of the plurality of instructions may be written to the worklist a predetermined number of times, wherein the predetermined number of times is based on the number of components in the superword register.

8. A method for static single assignment form dead code elimination comprising:

receiving a plurality of instructions;

adding to each instruction a previous bit;

adding to each instruction a write mask; and

generating a worklist by:

for each of the plurality of instructions; determining if the instruction is a critical instruction; and

if the instruction is a critical instruction, writing the instructions to the worklist.

9. The method of claim 8 further comprising:

examining a first instruction off of the worklist;

examining at least one second instruction in the machine code, wherein the at least one second instructions are sources of the first instruction;

determining if all elements within a particular field are live for the at least one second instruction; and

if no elements are live, deleting the first instruction from the worklist.

10. The method of claim 9 comprising:

prior to examining the first instruction off of the worklist, setting a live bit for each component of the plurality of instructions on the worklist.

11. The method of claim 8 wherein the critical instruction is an instruction that generates an export value.

12. The method of claim 8 wherein the write mask is a multi-bit field representing a number of components in a superword register.

13. The method of claim 12 wherein each of the plurality of instructions may be written to the worklist a predetermined number of times, wherein the predetermined number of times is based on the number of components in the superword register.

14. An apparatus for static single assignment form dead code eliminations comprising:

at least one memory device storing a plurality of executable instructions; and

at least one processor operably coupled to the at least one memory device, operative to receive the plurality of executable instructions such that the processor, in response to the executable instructions:

examines a first instruction off of a worklist, wherein the first instruction includes previous bit and a write mask;

examines at least one second instruction of the machine code, wherein the at least one second instructions are sources of the first instruction and each of the at least one second instructions include a previous link and a write mask;

determines if all components within a particular field are live for the at least one second instruction; and

if no elements are live, deletes the first instruction from the machine code.

15. The apparatus of claim 14 wherein the at least one processor further in response to the executable instructions:

generates the worklist by:

for each of a plurality of instructions, determining if the instruction is a critical instruction; and

if the instruction is a critical instruction, writing the instruction to the worklist.

sets a live bit for each component of the plurality of instructions.

16. The apparatus of claim 15 wherein the critical instruction is an instruction that generates an export value.

17. The apparatus of claim 15, wherein the at least one processor further in response to the executable instructions:

prior to generating the worklist:

receives a plurality of instructions;

adds to each instruction a previous link; and

adds to each instruction a write mask.

18. The apparatus of claim 17 further comprising:

a superword register operably coupled to the at least one processor, wherein the write mask is a multi-bit field representing a number of components in the superword register.

19. An apparatus for static single assignment form dead code eliminations comprising:

at least one memory device storing a plurality of executable instructions; and

at least one processor operably coupled to the at least one memory device, operative to

receive the plurality of executable instructions such that the processor, in response to the executable instructions:

receives a plurality of instructions;

adds to each instruction a previous link;

adds to each instruction a write mask; and

generates a worklist by:

for each of the plurality of instructions; determining if the instruction is a critical instruction; and

if the instruction is a critical instruction, writing the instructions to the worklist.

examines a first instruction off of the worklist;

examines at least one second instruction from the machine code, wherein the at least one second instructions are sources of the first instruction;

determines if any elements within a particular field are live for the at least one second instruction; and

if no elements are live, deletes the first instruction from the machine code.

20. The apparatus of claim 19 further comprising:

a superword register operably coupled to the at least one processor, wherein the write mask is a multi-bit field representing a number of components in a superword register.